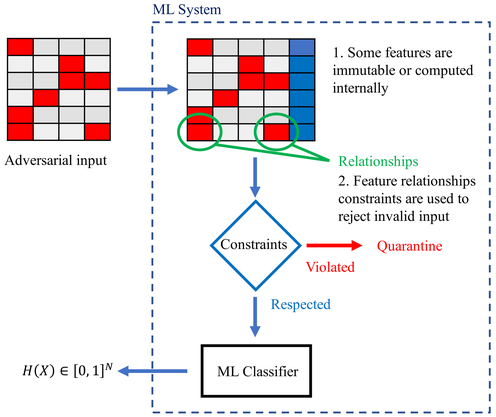
TabularBench: Benchmarking Adversarial Robustness for Tabular Deep Learning in Real-world Use-cases

Thibault Simonetto[1], Salah Ghamizi[2,3] and Maxime Cordy[1]
[1]University of Luxembourg, [2]Luxembourg institute of Science and Technologies, [3]Riken AIP
Contact: thibault.simonetto@uni.lu

Problem



Attacker objective

For x find δ such that:
- ✓ $H(x) \neq H(x+\delta)$
- ✓ $L_p(x, x+\delta) < \epsilon$
- ✓ $x + \delta \in \mathcal{X}_\Omega$

With:
$H: \mathcal{X} \rightarrow \mathcal{Y}$ the classification model
L_p the distance according to a $p-norm$
$\mathcal{X}_\Omega \subseteq \mathcal{X}$ the subspace where x satisfies all constraints $\omega \in \Omega$

Example of constraint:

$$installment = loan\_amount \times \frac{int\_rate \times (1+int\_rate)^{term}}{(1+int\_rate)^{term}-1}$$

Constraints grammar:

$$\omega := \omega_1 \wedge \omega_2 \mid \omega_1 \vee \omega_2 \mid \psi_1 \succeq \psi_2$$
$$\psi := c \mid f_i \mid \psi_1 \oplus \psi_2 \mid x_i$$

Known attacks produce invalid examples:

PGD ✗

AutoAttack ✗

	Max trans.	Avg trans.	Acc. creation	Age
Adversarial	$2500	$3000	1 year	22

Adversarial violates Avg $transaction \leq Max\ transaction$

Constraints as a penalty function:

Constraint	Penalty function	Constraint example	Penalty function example
$\psi_1 = \psi_2$	$\mid \psi_1 - \psi_2 \mid$	rec_per_month = record/month	$\mid$ rec_per_month $-$ (record/month) $\mid$
$\psi_1 \leq \psi_2$	$max(0, \psi_1 - \psi_2)$	open_acc $\leq$ total_acc	$max(0,$ open_acc $-$ total_acc$)$
$\psi_1 < \psi_2$	$max(0, \psi_1 - \psi_2 + \tau)$	open_acc $<$ total_acc	$max(0,$ open_acc $-$ total_acc $+10^{-5})$
$\omega_1 \wedge \omega_2$	$\omega_1 + \omega_2$	((term = 36) $\vee$ (term = 60)) $\wedge$ (open_acc $\leq$ total_acc)	min($\mid$ term $-$ 36 $\mid$, $\mid$ term $-$ 60 $\mid$) + $max(0,$ open_acc $-$ total_acc$)$
$\omega_1 \vee \omega_2$	$\min(\omega_1, \omega_2)$	(term = 36) $\vee$ (term = 60)	min($\mid$ term $-$ 36 $\mid$, $\mid$ term $-$ 60 $\mid$)

Tabular attacks

CAPGD (white-box)

Projected Gradient Descent (PGD)

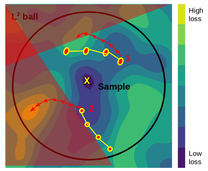
$$x^{(k+1)} = P_{\mathcal{S}}(x^{(k)} + \eta \nabla l(h(x), y))$$

Gradient loss + Constraints regularization

$$\mathcal{L}'(x) = l(h(x), y) - \sum_{\omega_i \in \Omega} penalty(x, \omega_i)$$

Constrained Gradient Descent

$$z^{(k+1)} = P_{\mathcal{S}}(x^{(k)} + \eta^{(k)}(\nabla \mathcal{L}'(x^{(k)}))$$



Repair operator R_Ω

$$\omega_1 \equiv installment = loan\_amount \times \frac{int\_rate \times (1+int\_rate)^{term}}{(1+int\_rate)^{term}-1}$$

Constrained Adaptive Attack (CAA)

Sequential attack: CAPGD → MOEVA

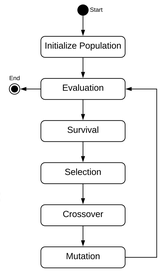
Efficiency

Effectiveness

MOEVA (gray-box)

Multi-objective genetic algorithm (NSGA-III)

$$minimise\ g_1(x) \equiv h(x)$$
$$minimise\ g_2(x) \equiv L_p(x - x_0)$$
$$minimise\ g_3(x) \equiv \sum_{\omega_i \in \Omega} penalty(x, \omega_i)$$



Metrics

Adversarial *examples* = ***Perturbed*** *examples if satisfied constraints*
Revert to ***Original*** *examples otherwise*

$$\textbf{\textit{Robust}}\ accuracy = \frac{\#\ Correctly\ classified\ \textbf{\textit{adversarial}}\ examples}{\#\ Clean\ examples}$$

TabularBench

Regularized adversarial training

$$\mathcal{L}_B = \frac{1}{(m-k)+\lambda k}\left(\lambda \sum_{i=1}^{k} \mathcal{L}(h(PGD(x_i, y_i)), y_i) + \sum_{i=k+1}^{m} \mathcal{L}(h(x_i), y_i)\right)$$

Regularized adversarial training with data augmentation

$$\mathcal{L}_B = \frac{1}{\sum_{j=1}^{4} \lambda_j k_j}\Big($$
$$\lambda_1 \sum_{i=1}^{k_1} \mathcal{L}(h(x_i, y_i), y_i) + \quad \text{Clean examples}$$
$$\lambda_2 \sum_{i=k_1+1}^{k_1+k_2} \mathcal{L}(h(PGD(x_i, y_i)), y_i) + \quad \text{Adversarial examples}$$
$$\lambda_3 \sum_{i=k_1+k_2+1}^{k_1+k_2+k_3} \mathcal{L}(h(GEN(x_i, y_i)), y_i) + \quad \text{Synthetic examples}$$
$$\lambda_4 \sum_{i=k_1+k_2+k_3+1}^{m} \mathcal{L}(h(PGD(GEN(x_i, y_i), y_i)), y_i)\Big) \quad \text{Synthetic and Adversarial examples}$$

Training

- Standard
- Adversarial training

Data augmentation

- None
- CT-GAN
- GOGGLE
- TableGAN
- TVAE
- WGAN
- CutMix

Model architectures

- RLN
- STG
- TabNet
- TabTransformer
- VIME

Three domains, five datasets

Credit scoring Lending Club Loan Data (LCLD)

Botnet detection CTU

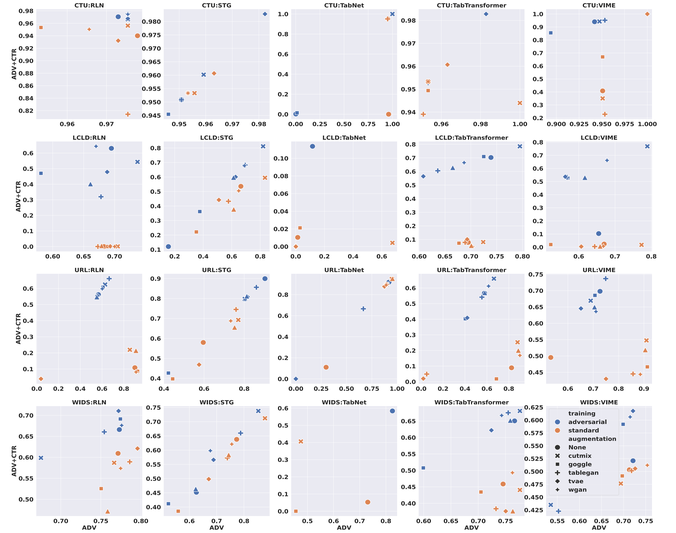
URL phishing URL

Malware detection MALWARE

ICU survival WIDS

290 Pre-trained models

Results



Robust performance while considering domain constraints (ADV+CTR: Y-axis) and without (ADV: X-axis) on all our use cases confirms the relevance of studying constrained-aware attacks.

API

Available on PyPi, Conda and Docker

```
pip install tabularbench
```

Models on Hugging Face

🤗 serval-uni-lu/tabularbench

Access to datasets and models ↓

```python
from tabularbench.models.tab_scaler import TabScaler
from tabularbench.datasets.dataset_factory import get_dataset
from tabularbench.benchmark.model_utils import load_model_and_weights

dataset = "url"
model_arch = "stg"
model_training= "default"
ds = get_dataset(dataset)
metadata = ds.get_metadata(only_x=True)
scaler = TabScaler(num_scaler="min_max", one_hot_encode=True)

model_eval = load_model_and_weights(
    ds.name, model_arch, model_training, metadata, scaler
)
```

Constraints access and definition ↓

```python
from tabularbench.constraints.relation_constraint import Feature, Constant
from tabularbench.datasets.samples.lcld import get_relation_constraints

lcld_constraints = get_relation_constraints()

new_constraint = (
    (Feature("term") == Constant(36)) |
    (Feature("term") == Constant(48)) |
    (Feature("term") == Constant(60))
)
lcld_constraints[3] = new_constraint
```

Learn more

Paper

Code

This project is supported by the Luxembourg National Research Fund and BGL BNP Paribas Luxembourg.